TOYOTA TECHNOLOGICAL INSTITUTE AT CHICAGO

Grammar Induction from Natural Supervision

Haoyue Shi freda@ttic.edu
Toyota Technological Institute at Chicago
July 26th, 2020

Background: Basic concepts

Supervised learning: train on A, test on A.

Distantly supervised learning: train on A, test on B.

Unsupervised learning: train on no (manually collected) labels;
sometimes confused with self-supervised learning.

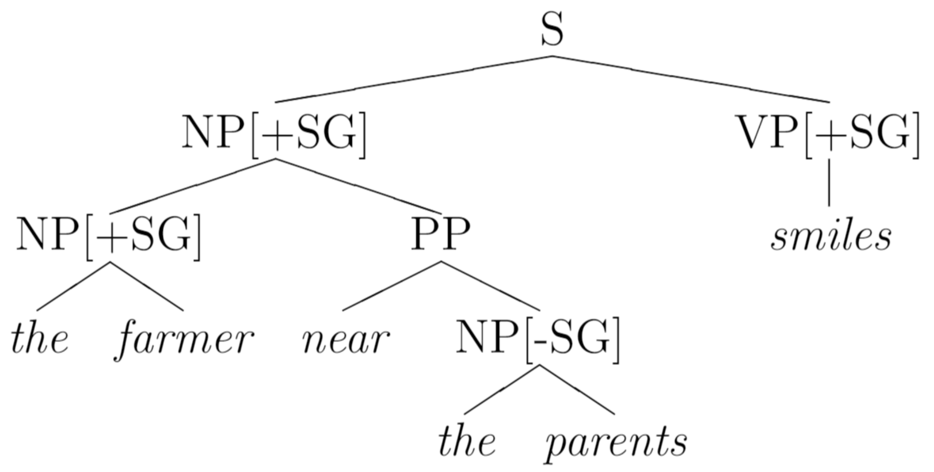
Natural supervision: labels that can be acquired “naturally”;
sometimes confused with self supervision.

Self-supervised learning: Not to be confused with self-training/self-learning.
Anyway, we don’t want to put so much effort on these concepts.

Which one is acceptable?

(1) The farmer near the parents smile. *

(2) The farmer near the parents smiles.



(3) The farmer that the parents love swim. *

(4) The farmer that the parents love swims.

[Example credit: Marvin and Linzen, 2018]

Language has structure

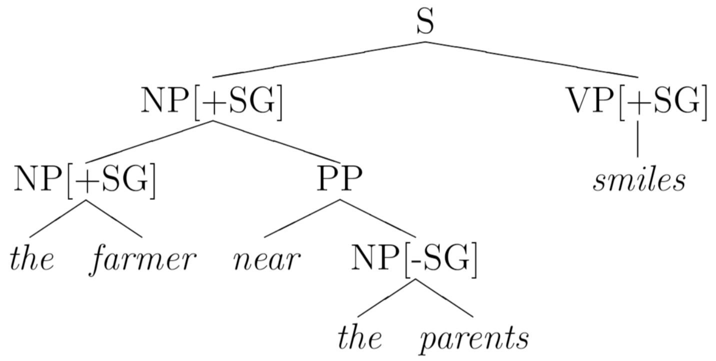
Humans learn language efficiently and effectively.

We implicitly develop and use structure for (natural) language processing.

Such structure is almost never explicitly shown.

Question: can we build a model which induces natural language structure ***naturally***?

In this talk, I will use grammar induction and unsupervised parsing interchangeably.



Recent work on unsupervised parsing

Input: a large set of sentences. Output: the induced parse trees.

Optional input: Labels for other tasks.

Some representative work:

DIORA (Drozdov et al., NAACL 19): https://www.aclweb.org/anthology/N19-1116/

URNNG (Kim et al., NAACL 19): https://www.aclweb.org/anthology/N19-1114/

Depth-Bounded PCFG (Jin et al., TACL 18): https://www.aclweb.org/anthology/Q18-1016.pdf

Compound PCFG (Kim et al., ACL 19): https://www.aclweb.org/anthology/P19-1228/

Distantly supervised parsing (Li et al., ACL 19): https://www.aclweb.org/anthology/P19-1338/

How did we learn our (first) language?

A cat is on the lawn.

How did we learn our (first) language?

A cat was chasing a mouse.
A dog was chasing a cat.
A cat was chased by a dog.
…

A cat is on the lawn.
A cat is staring at you.
A cat plays with a ball.

A cat, as a whole,
means something concrete.

A cat sleeps outside.
A cat is on the ground.
There is a cat sleeping on the ground.

A cat, as a whole, functions as a single unit in sentences.

Problem definition

Learning to induce language structure from natural supervision: Given a large set of parallel image-text data (e.g., MS-COCO), can we generate linguistically plausible structure for the text?

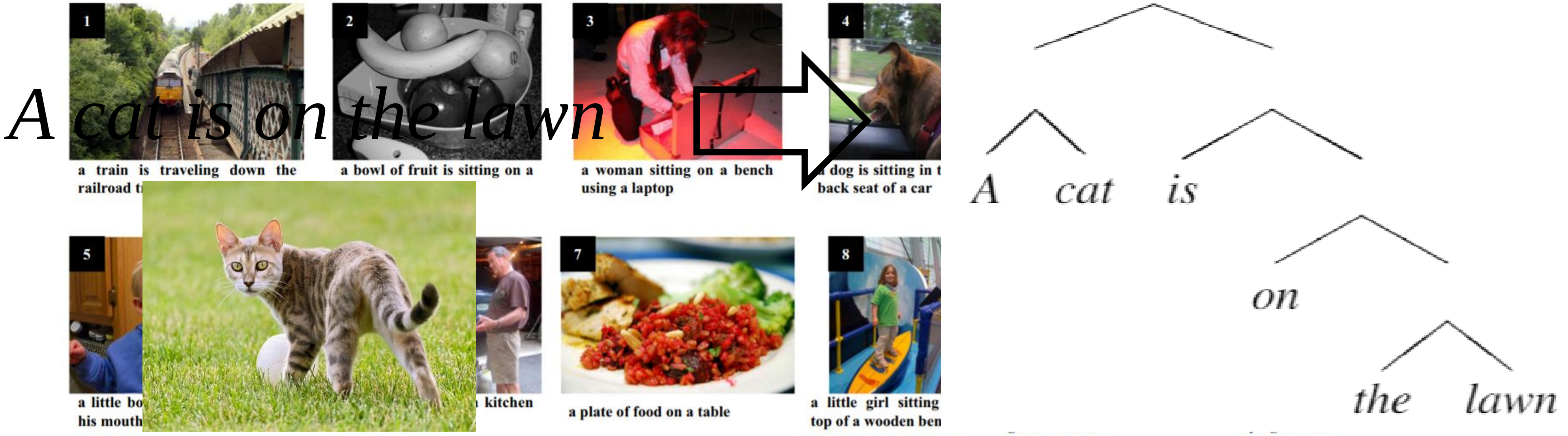
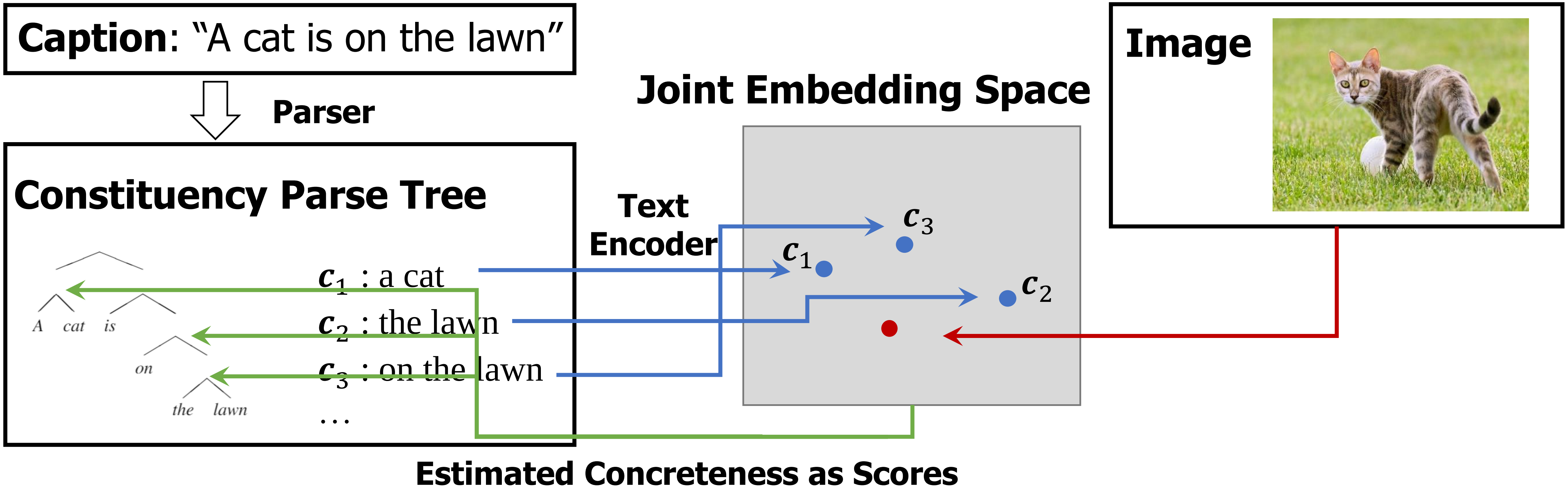


Figure credit: Ding et al. (2018)

The Visually Grounded Neural Syntax Learner

Basic assumption: *Concrete* spans are more likely to be constituents.



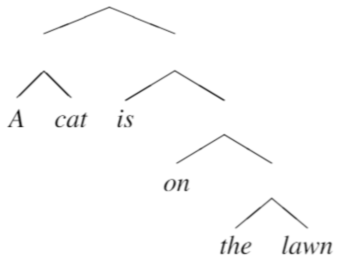
The Visually Grounded Neural Syntax Learner

Basic assumption: *Concrete* spans are more likely to be constituents.

Caption: "A cat is on the lawn"

Parser

Constituency Parse Tree



$\boldsymbol{c}_1$: a cat
$\boldsymbol{c}_2$: the lawn
$\boldsymbol{c}_3$: on the lawn
…

Greedy Bottom-Up Parser

a cat is on the lawn

Greedy Bottom-Up Parser

Compute score

$$FFN\left(\begin{bmatrix}\mathbf{v}_a \\ \mathbf{v}_{cat}\end{bmatrix}\right) = 4.5$$

4.5

a cat is on the lawn

Greedy Bottom-Up Parser

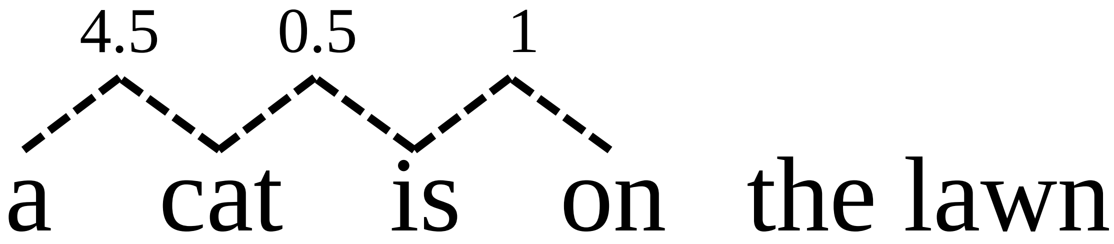
Compute score

$$FFN\left(\begin{bmatrix}\mathbf{v}_{cat}\\ \mathbf{v}_{is}\end{bmatrix}\right) = 0.5$$

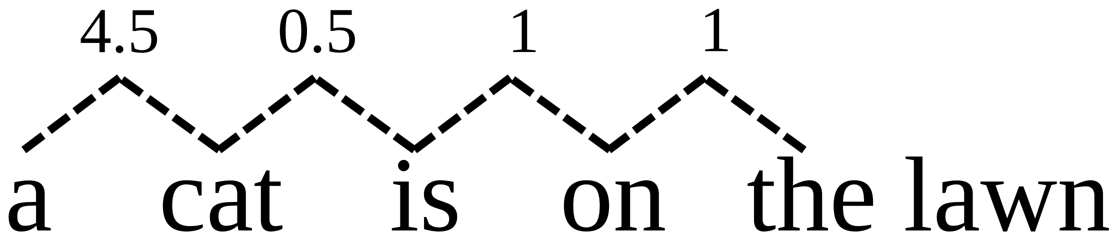
Greedy Bottom-Up Parser

Compute score

$$FFN\left(\begin{bmatrix}\mathbf{v}_{is}\\ \mathbf{v}_{on}\end{bmatrix}\right) = 1$$



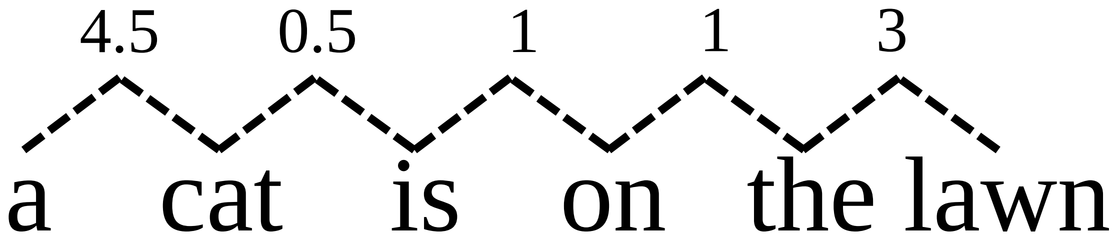
Greedy Bottom-Up Parser



Compute score

$$FFN\left(\begin{bmatrix}\mathbf{v}_{on}\\ \mathbf{v}_{the}\end{bmatrix}\right) = 1$$

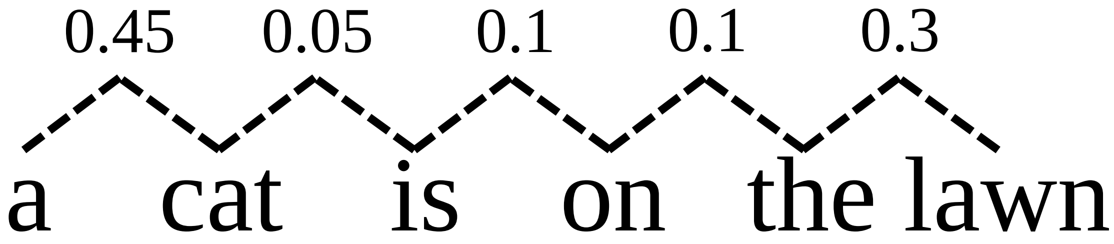
Greedy Bottom-Up Parser



Compute score

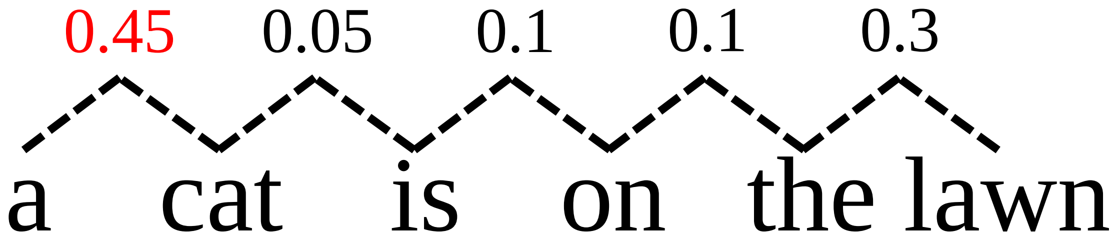
$$FFN\left(\begin{bmatrix}\mathbf{v}_{the}\\ \mathbf{v}_{lawn}\end{bmatrix}\right) = 3$$

Greedy Bottom-Up Parser



Normalized to a probability distribution

Greedy Bottom-Up Parser



Sample a pair to combine (training)
Greedily combine (inference)

Greedy Bottom-Up Parser

(a cat)

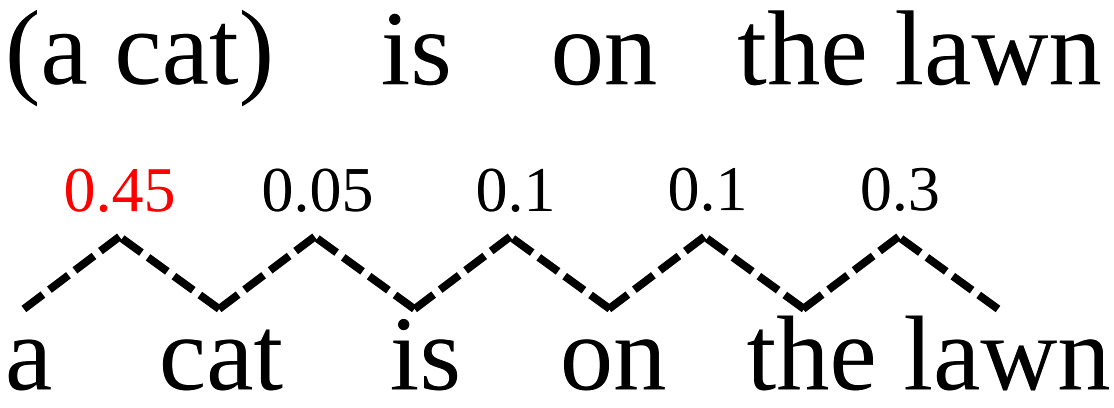
0.45 0.05 0.1 0.1 0.3

a cat is on the lawn

Textual representation:
Normalized sum of children

$$\mathbf{v}_{(a\ cat)} = \frac{\mathbf{v}_a + \mathbf{v}_{cat}}{||\mathbf{v}_a + \mathbf{v}_{cat}||_2}$$

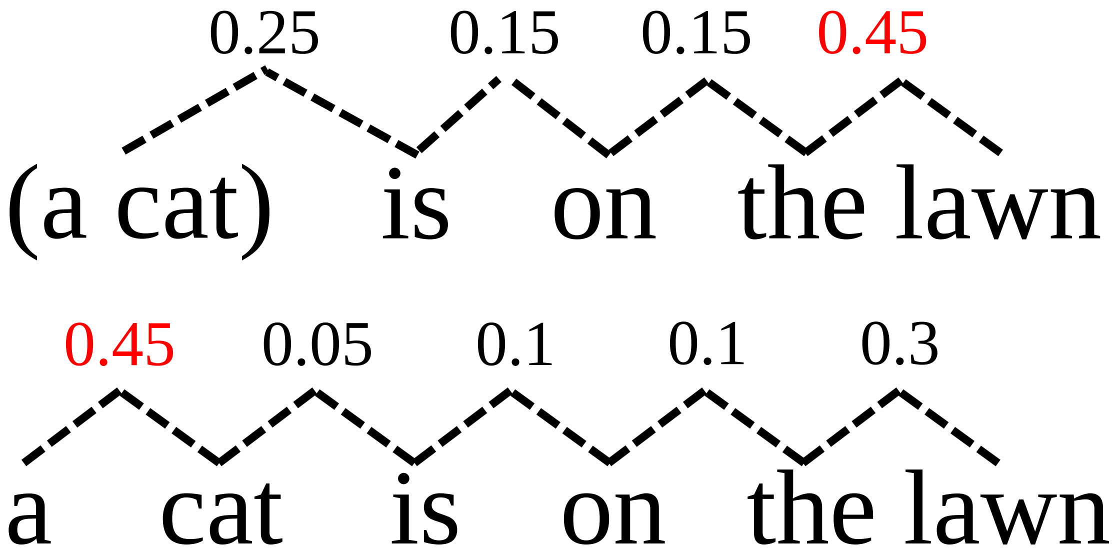
Greedy Bottom-Up Parser



Textual representation:
Normalized sum of children

$$\mathbf{v}_{(a\ cat)} = \frac{\mathbf{v}_a + \mathbf{v}_{cat}}{\left|\left|\mathbf{v}_a + \mathbf{v}_{cat}\right|\right|_2}$$

Greedy Bottom-Up Parser



Compute probability

Greedy Bottom-Up Parser

(a cat) is on (the lawn)

Combine

0.25 0.15 0.15 0.45

(a cat) is on the lawn

0.45 0.05 0.1 0.1 0.3

a cat is on the lawn

Greedy Bottom-Up Parser

((a cat) (is (on (the lawn))))

Finished!

. . .

(a cat) is on (the lawn)

0.25 0.15 0.15 0.45

(a cat) is on the lawn

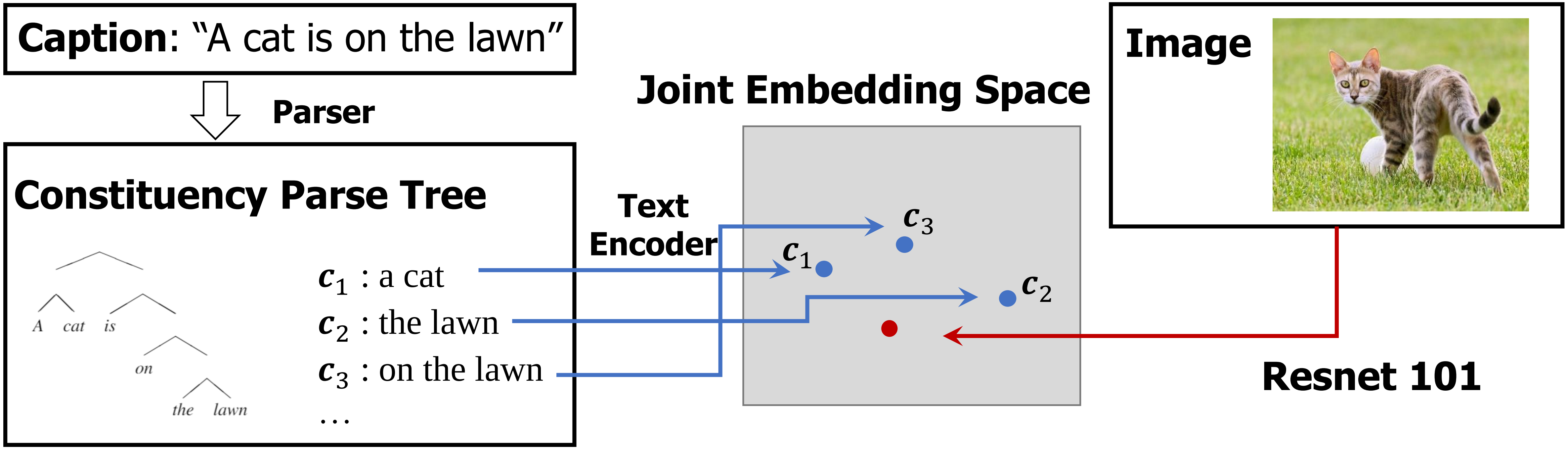
(Trainable) parameters:
The parameters of the scoring FFN.

0.45 0.05 0.1 0.1 0.3

a cat is on the lawn

The Visually Grounded Neural Syntax Learner

Basic assumption: *Concrete* spans are more likely to be constituents.



The Joint Embedding Space

Hinge-based triplet loss between images and captions for visual semantic embeddings (VSE; Kiros et al., 2015):

A cat is on the lawn.

A dog is on the lawn.

$$\mathcal{L}(i,c) = \sum_{(i',c')\neq(i,c)} [sim(i',c) - sim(i,c) + \delta]_+ + [sim(i,c') - sim(i,c) + \delta]_+$$

A cat is on the lawn.

$$[\cdot]_+ = \max(\cdot, 0)$$
$$sim(\cdot,\cdot) = \cos(\cdot,\cdot)$$

Concreteness Estimation in the Joint Embedding Space

a cat √

on the ?

Hinge-based triplet loss between images and ~~captions~~ constituents for visual semantic embeddings:

$$\mathcal{L}(i,c) = \sum_{(i',c') \neq (i,c)} [sim(i',c) - sim(i,c) + \delta]_+ + [sim(i,c') - sim(i,c) + \delta]_+$$

Abstractness: local hinge loss between constituents and images.

$$abstract(c;i) = \mathcal{L}(i,c)$$

Concreteness is defined similarly:

$$concrete(c;i) = \sum_{(i',c') \neq (i,c)} [-sim(i',c) + sim(i,c) - \delta]_+ + [-sim(i,c') + sim(i,c) - \delta]_+$$

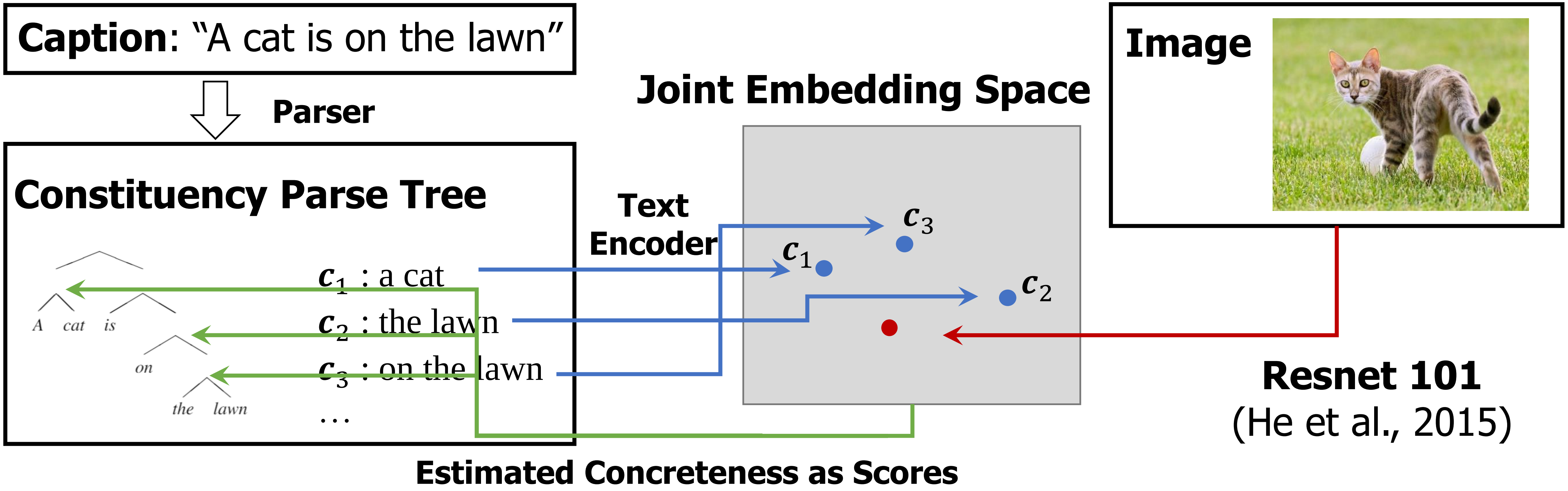
$$[\cdot]_+ = \max(\cdot, 0)$$

$$sim(\cdot,\cdot) = \cos(\cdot,\cdot)$$

The Visually Grounded Neural Syntax Learner

Basic assumption: *Concrete* spans are more likely to be constituents.

REINFORCE (Williams, 1992) as gradient estimator for parser.



Where should function words go?

((A cat) on) (the lawn) (A cat) (on (the lawn)) √

Fact #1: *On* is the head of *on the lawn*.

Fact #2: English is strongly head-initial.
Many other Indo-European languages are head-initial as well.

Fact #3: Under the setting of visual grounding, most abstract words are function words (e.g., prepositions, determiners, complementizers).

Empirical Solution (mimic the head-initial property):
Discourage abstract words from combining to the front.

$$c = [c_{left}; c_{right}]$$
$$reward(c) = concrete(c; i)$$

⇒

$$reward(c) = \frac{concrete(c; i)}{\lambda \cdot abstract(c_{right}; i) + 1}, \quad \lambda > 0$$

Training and Evaluation

Datasets	Language	# Image (train/dev/test)	# Caption (train/dev/test)
MSCOCO (Lin et al., 2014)	EN	80K/1K/1K	400K/5K/5K
Multi30K (Elliott et al., 2016)	EN, DE, FR	28K/1K/1K	28K/1K/1K

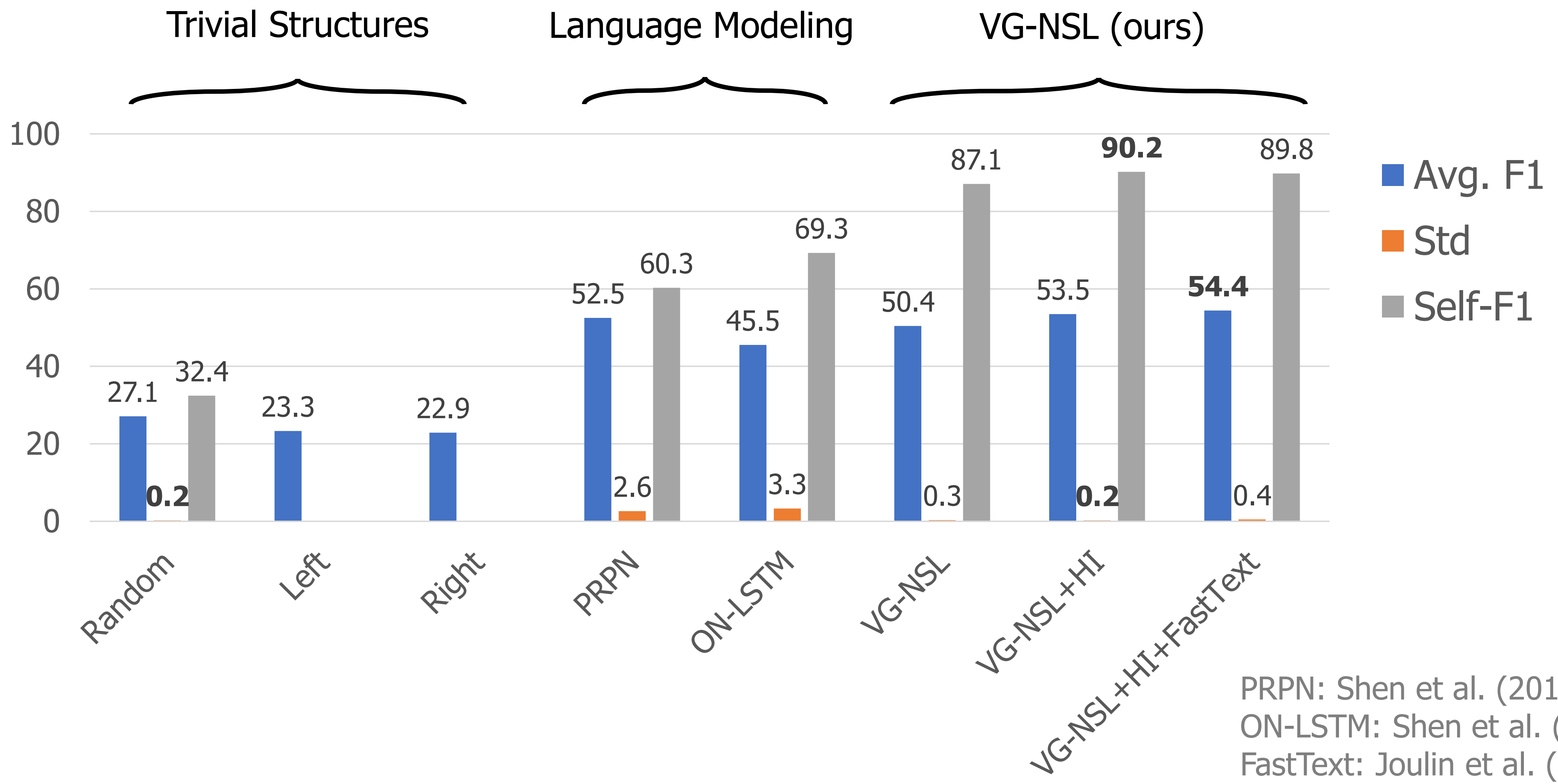
Each model takes 5 runs, with different random seeds

F_1: Average agreement with Benepar (Kitaev and Klein, 2018)

Std: Standard deviation of F_1 scores

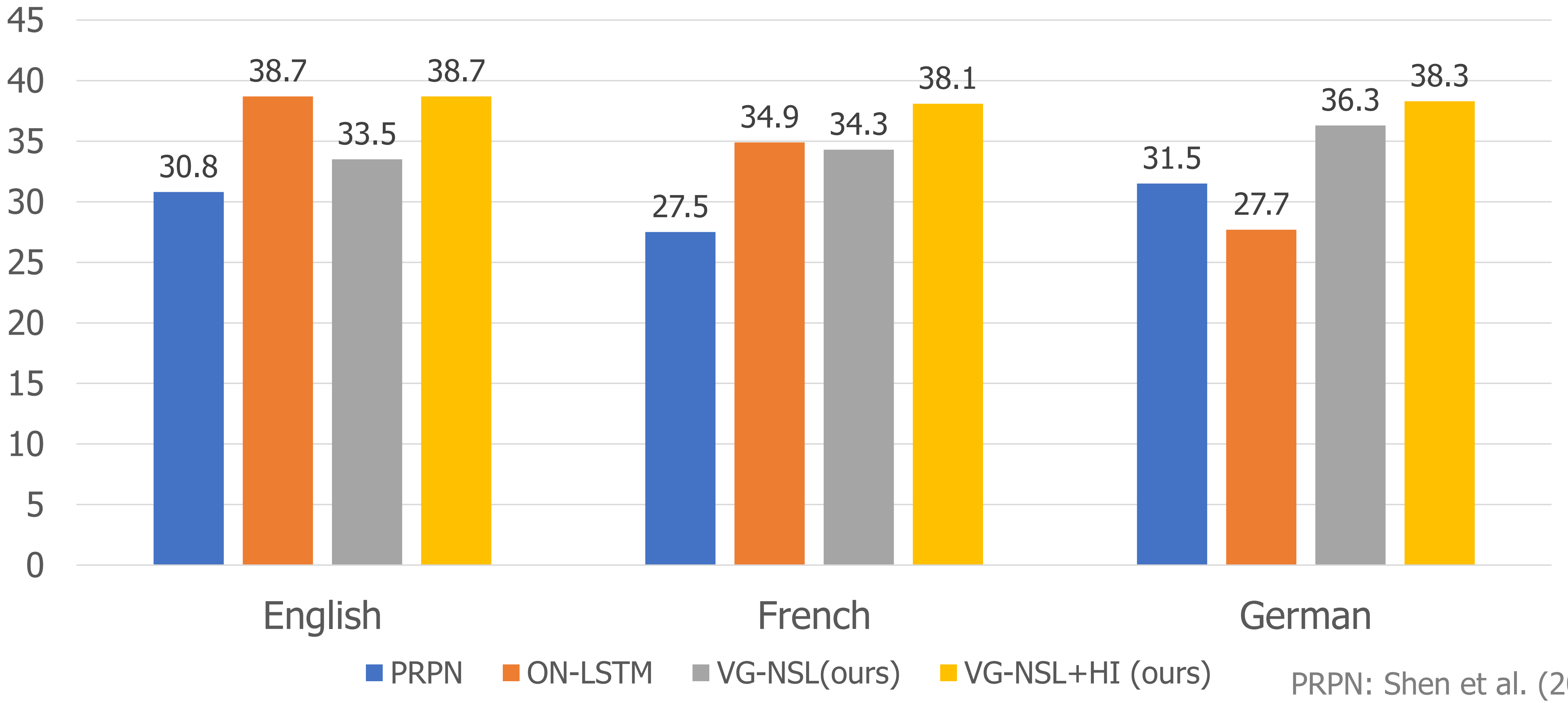
Self-F_1: Average agreement across the $\binom{5}{2}$ pairs of models

Unsupervised/Naturally Supervised Parsing



PRPN: Shen et al. (2018)
ON-LSTM: Shen et al. (2019)
FastText: Joulin et al. (2016)

Performance on Multiple Languages



PRPN: Shen et al. (2018)
ON-LSTM: Shen et al. (2019)

Limitations and potential future direction

It only works on concrete domains. (Kojima et al., ACL 2020)

How can we transfer it to other domains?

Embedding alignment.

A book on **the desk**

A book on **the topic of philosophy**

How about head-final languages? Can we learn head-directionality?

Thoughts about Grammar Induction

Does mutual information work?   PMI: 30.5   VGNSL-HI: 53.3   Random: 27.1

Klein and Manning (2004): No.

This below-random performance seems to be because the model links word pairs which have high mutual information (such as occurrences of *congress* and *bill*) regardless of whether they are plausibly syntactically related.

Mutual information reflects a mixture of syntax and semantics.

Most self-supervised signals from pure text are about mutual information.

We should keep this in mind when designing new GI models.

Thoughts about Grammar Induction

Cho (2018), Kim et al. (2019): Tuning on a labeled development set is *not* fully unsupervised parsing.

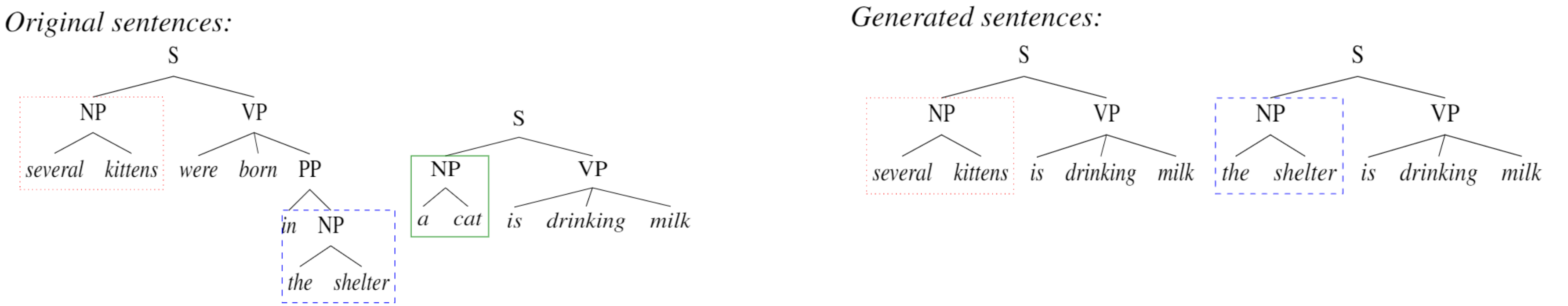
No matter how we used the label, we used them.

If tuning on a labeled development set, we should consider

1. Using as few labeled sentences as possible;

2. Comparing to a strong baseline: **training** with the same set.

Few-shot parsing as the baseline



Many unsupervised parsing models are tuned w.r.t. F1 score on WSJ dev set (1,700 sentences). Among them, DIORA (Drozdov et al., 2019) is the best.

Model	Test F1
DIORA (Drozdov et al., 2019)	56.5
FSP (|Train|=50, |Dev|=5)	57.5
FSP + SUB	**79.4**

[Supervised parser: Benepar (Kitaev and Klein, 2018)]

Thoughts about Grammar Induction

Williams et al. (2018): Random seeds matter in terms of tree structure, but they do not matter in terms of downstream performance.

Models that can be affected a lot by random seeds are not desired.

Unfortunately, many unsupervised parsing or latent tree models are sensitive to random seeds.

Question: Are there multiple optimal structures for downstream tasks?

Are the optimal structures linguistically plausible?

[Williams, A., Drozdov, A., Bowman, S.R. (2018). Do latent tree learning models identify meaningful structure in sentences? In TACL.]

Prior Study

Are there multiple optimal structures for downstream tasks?

Williams et al. (2018): Yes, for natural language inference.

Are the optimal structures linguistically plausible?

Williams et al. (2018): No.

What if we train on downstream tasks, with some linguistic regularizations?

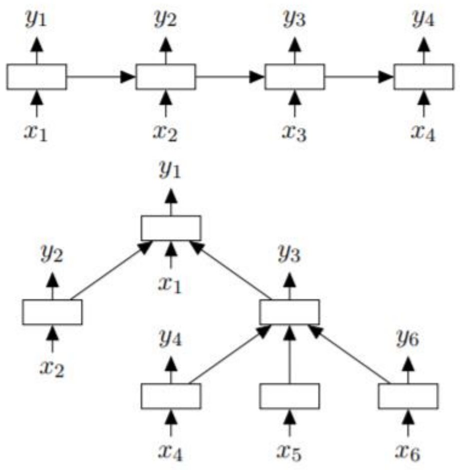
$$\mathcal{L} = \mathcal{L}_{downstream} + \lambda \mathcal{L}_{parse} \quad (\lambda \ll 1)$$

Similar phenomenon to that reported by Williams et al. (2018).

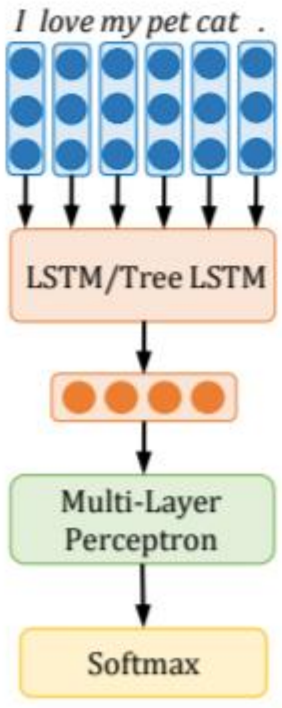
[Williams, A., Drozdov, A., Bowman, S.R. (2018). Do latent tree learning models identify meaningful structure in sentences? In TACL.]

Tree based neural sentence modeling

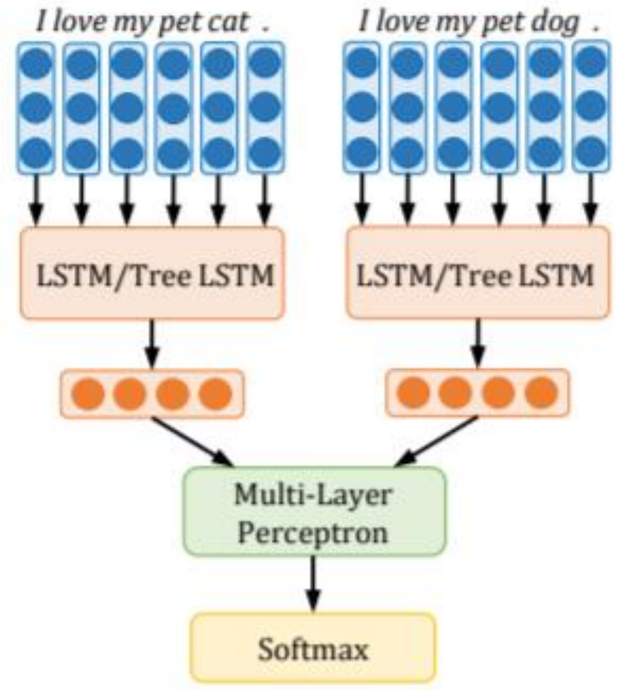
Constituency parse tree based Tree LSTMs (Zhu et al., 2015; Tai et al., 2015) was popular for sentence modeling.



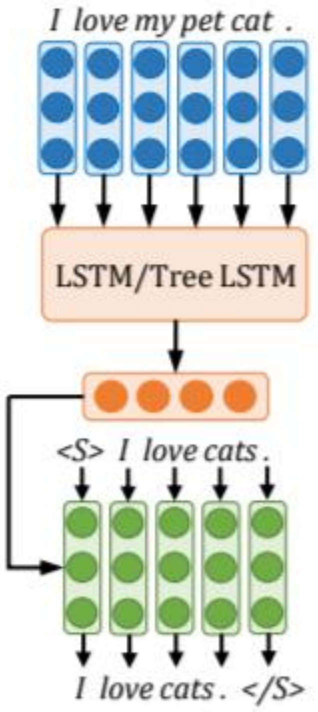
[Figure credit: Tai et al., 2015]



Sentence classification



Sentence relation classification



Sentence generation

Tree based neural sentence modeling

Constituency parse tree based Tree LSTMs (Zhu et al., 2015; Tai et al., 2015) was popular for sentence modeling.

Different constituency-style tree can be used to substitute the parse tree.

(a) Parsing tree. (b) Balanced tree. (c) Gumbel tree. (d) Left-branching tree. (e) Right-branching tree.

Gumbel tree (Choi et al., 2018): a latent tree model which enables backpropagation through discrete structures.

[Choi, J., Yoo, K. M., Lee, S-g. (2018) Learning to compose task-specific tree structures. In AAAI]

Tree based neural sentence modeling

Considered downstream tasks

Sentence classification:
AG's news (topic)
Amazon review (sentiment)
DBpedia (topic)
word-level semantic relation

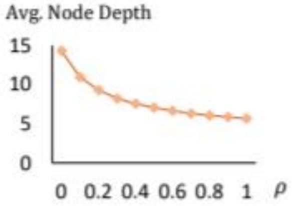
Sentence relation:
Natural language inference
Conjunction prediction

(Conditioned) sentence generation:
Autoencoding
Paraphrasing
Machine translation

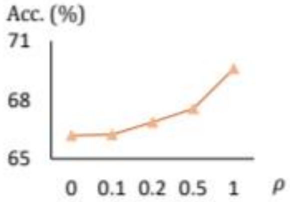
Tree based neural sentence modeling

	Sentence Classification					*Sentence Relation*		*Sentence Generation*		
Model	**AGN**	**ARP**	**ARF**	**DBpedia**	**WSR**	**NLI**	**Conj**	**Para**	**MT**	**AE**
Latent Trees										
Gumbel	91.8	87.1	48.4	98.6	66.7	80.4	51.2	20.4	17.4	39.5
+bi-leaf-RNN	91.8	**88.1**	**49.7**	98.7	69.2	**82.9**	53.7	20.5	22.3	75.3
(Constituency) Parsing Trees										
Parsing	91.9	87.5	49.4	**98.8**	66.6	81.3	52.4	19.9	19.1	44.3
+bi-leaf-RNN	92.0	88.0	49.6	**98.8**	68.6	82.8	53.4	20.4	22.2	72.9
Trivial Trees										
Balanced	92.0	87.7	49.1	98.7	66.2	81.1	52.1	19.7	19.0	49.4
+bi-leaf-RNN	**92.1**	87.8	**49.7**	**98.8**	**69.6**	82.6	**54.0**	20.5	22.3	76.0
Left-branching	91.9	87.6	48.5	98.7	67.8	81.3	50.9	19.9	19.2	48.0
+bi-leaf-RNN	91.2	87.6	48.9	98.6	67.7	82.8	53.3	20.6	21.6	72.9
Right-branching	91.9	87.7	49.0	**98.8**	68.6	81.0	51.3	20.4	19.7	54.7
+bi-leaf-RNN	91.9	87.9	49.4	98.7	68.7	82.8	53.5	**20.9**	**23.1**	**80.4**
Linear Structures										
LSTM	91.7	87.8	48.8	98.6	66.1	82.6	52.8	20.3	19.1	46.9
+bidirectional	91.7	87.8	49.2	98.7	67.4	82.8	53.3	20.2	21.3	67.0
Avg. Length	31.5	33.7	33.8	20.1	23.1	11.2	23.3	10.2	34.1	34.1

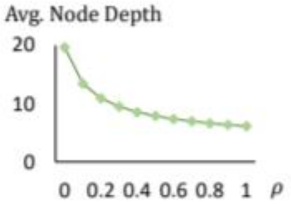
Why trivial structures work?



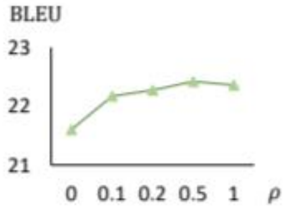
(a) ρ-depth line for WSR.



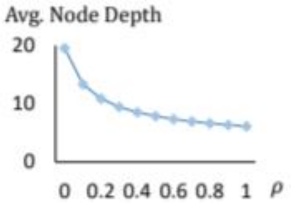
(b) ρ-Acc. line for WSR.



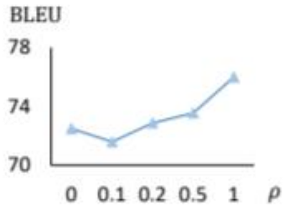
(c) ρ-depth line for MT.



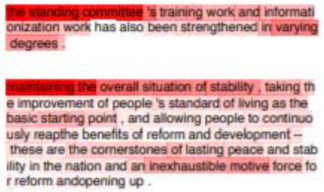
(d) ρ-BLEU line for MT.



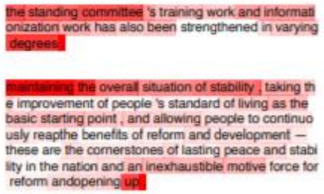
(e) ρ-depth line for AE.



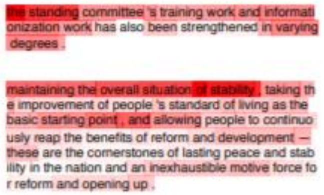
(f) ρ-BLEU line for AE.



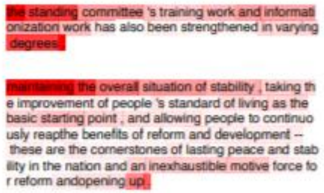
(a) Balanced tree, MT.



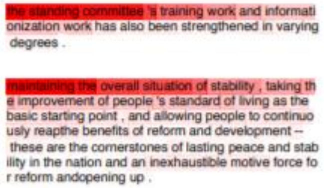
(b) Left-branching tree, MT.



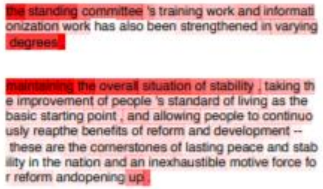
(c) Right-branching, MT.



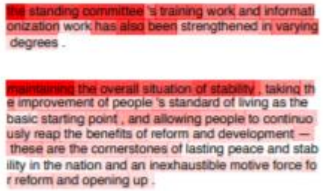
(d) Bi-LSTM, MT.



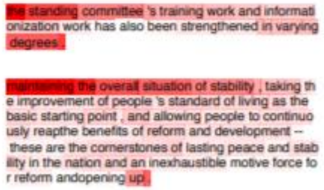
(e) Balanced tree, AE.



(f) Left-branching tree, AE.



(g) Right-branching, AE.



(h) Bi-LSTM, AE.

$$J(\mathbf{s}, \mathbf{w}) = \|\nabla \mathbf{s}(\mathbf{w})\|_1 = \sum_{i,j} \left| \frac{\partial s_i}{\partial w_j} \right|$$

Larger ρ means more balanced structure.

Tree based neural sentence modeling

Conclusions

- Tree based methods give better results when crucial words are closer to the final representation.
- For most real (semantic) downstream tasks, parse tree is not the best friend of Tree LSTMs.
- Consider using the Transformers!

Thoughts about Grammar Induction

Q: Does agreement with linguistic definition mean everything?

A: Not necessarily.

Q: Is there any downstream task that can benefit from induced parse tree?

A: Still an open question.

Q: Why doing unsupervised parsing?

A: Find statistical support for linguistic arguments.

Downstream tasks could probably benefit from the induced trees.

Constituent test

Acceptability (informal): does the sentence look good to you?

Example sentence: *Drunks could put off the customers in the bar.*

Coordination: [Drunks] <u>and bums</u> could put off the customers in the bar.

Substitution: <u>They</u> could put off the customers in the bar.

Topicalization: … and [put off the customers], drunks certainly could.

Deletion: *Drunks could put off the customers ~~in the bar~~.*

Can we computationally model constituent test?

Constituent test with GPT

GPT-2 (Radford et al., 2019) is a strong toolkit.

We can use GPT-2 LM score as the constituency test result, and select spans with high scores as constituent.

Challenges

The ball is on the table behind the sofa.

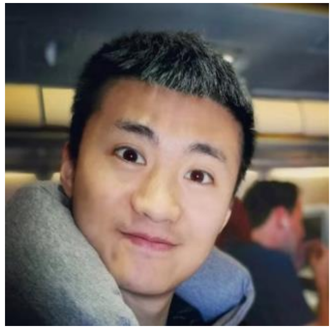
Deletion test: The ball is on ~~the table behind~~ the sofa.

Language models favor short sentences.

Thank you!

And, kudos to my collaborators!

Jiayuan Mao



Hao Zhou

Lei Li

Kevin Gimpel

Karen Livescu

智東西 公开课

专注讲解新兴技术创新与应用